

“Le rapport de projet chatbot”

“Shamim SEDGHI”

“UE programmation web 1”

“Abderrahman TLEMÇANI”

“Semestre 2 L1 Automne 2021”

Introduction :

Dans le cadre de l'UE programmation web de notre seconde semestre de L1 d'informatique à l'université de Strasbourg, il nous est demandé un projet de créer un chatbot nous permettant de mettre en pratique nos connaissances et nos compétences en programmation web concentré sur HTML, CSS, JAVASCRIPT. Notre objectif était de réaliser un terminal interactif (chatbot) dans lequel l'utilisateur aura la possibilité d'agir sur le CSS de la page à travers l'utilisation de commandes à implémenter.

Dans ce dossier nous allons tout d'abord exposer l'analyse technique du problème posé et enfin les solutions techniques proposées et justification des choix techniques effectués.

1. Analyse du sujet et problème posé

1.1 présentation du sujet

Le logiciel représente un terminal interactif en 2 versions logiciels, il s'agit d'une version mobile et une version de la page web sur navigateur. Pour entamer la description en détail de l'application nous détaillons les 2 versions séparément en nous concentrant sur la version page web.

1.1.2) agencement responsive :

A) la version page web(Desktop) : lorsque la page s'affiche sur navigateur , l'utilisateur devrait voir 2 parties principales :

- 1.** la partie de chat entre utilisateur et le bot (chat_zone) qui contient une barre d'écriture et la bouton de “ Send “ qui pourrait marcher soit par le click sur le bouton «send », soit avec la touche d'entrée du clavier. Ainsi la partie d'affichage des messages échangés entre le bot et l'utilisateur.
- 2.** la partie Help qui devrait être développée pour guider l'utilisateur afin de bien écrire les messages possibles pour recevoir les bonnes réponses de la part de la bot.

B) version Mobile : Elle devrait posséder tous les paramètres expliqués en version Desktop en ayant 2 différences avec la version page web :

1. La zone de message prend une grande partie de l'écran pour un meilleur affichage
2. Ajouter un bouton cliquable afin d'afficher la partie Help qui va apparaître à gauche de l'écran en cliquant sur le bouton.

2. les solutions techniques proposées

A) La version page web (Desktop) : Ainsi d'avoir expliqué auparavant nous créons 3 parties différentes : HTML , CSS , JAVASCRIPT .

1.HTML : Dans cette partie nous essayons de faire la structure générale du chatbot avec les ID et des classes différents et précis. Pour détailler et mieux comprendre les principes du codes , nous l'avons expliqué dans le fichier readme.

2.CSS : En général, nous avons créé du style sur les différentes classes du fichier HTML en précisant les paramétrage apparences comme : les couleurs, les fonts , la position de chaque partie , les tailles, et avec les différents aspects des flex. Comme indiqué dans le sujet, nous devons afficher le dernier message en format gras (bold), qu'il a été effectué dans la partie ".chat_zone.message" on indiquant le last child en gras. On a essayé au maximum d'éviter les potion fixe et Absolut pour que la page reste responsive au maximum.

3 .JAVASCRIPT : Pour la partie la plus importante qu'il s'agit de la partie JAVA SCRIPT , tout d'abord nous pourrons voir la dictionnaire créer afin de regrouper tout ce qui « fonction la réponse a cette fonction et l'aide ». "styleDejaAppliques " cette variable vas être utiliser dans la fonction reset (fonction qui permet de restaurer les styles de base), dans la fonction affichemesage vous pourriez remarquer que nous avons précisé les parties concernées par utilisateur par " True " et les parties concernées par le bot par "False" puisque on a juste deux état possible donc deux possibilité de traitement(le côté utilisateur ou bot dans .chat-zone). On a aussi ajouter la possibilité d'envoyer un message de l'utilisateur par le bouton entrée du clavier

B) La version Mobile : cette version inclut les mêmes codes de HTML sauf quelques précisions en CSS et le javascript.

1.CSS: Dans la partie “media screen” nous avons mis une position “absolute” c’est le moyen qu’on trouvé le plus facile, de plus que ça ne vas affecter la page vue qu’elle devrait être toujours en haut a droit de la page. Ainsi dans la même partie, “transition” de type “ ease -in-out” qui nous permettra d’avoir la partie Help qui va être affichée de gauche a droit tout en augmentant la vitesse.

2.javascript

La fonction on onClickHelpBtn qui permettra de show la partie aide ou de la caché dans la classe help.